

REMARKS

Claims 1-16 were presented for examination and were rejected. Reconsideration of this application in view of the following remarks, and allowance of all claims herein, claims 1-16 as written, are hereby respectfully requested.

In her second paragraph, the Examiner rejected claims 1-16 under 35 U.S.C. §102(e) as being anticipated by Chen. Applicant hereby traverses this rejection. All of Applicant's independent claims (claims 1, 13, 14, and 16), and thus all of Applicant's claims, contain two critical limitations that are not suggested by Chen:

1) the code contains instructions causing a macro to be moved to a global environment; and

2) the code contains instructions causing a macro to be copied to a local document.

The Examiner cited column 5, lines 3-14 and column 6, lines 38-60 of Chen for the first proposition (a macro is moved to a global environment). These passages of Chen do not suggest a macro being moved to a global environment. The portion of Chen most relevant to Applicant's first limitation is column 13 line 50 through column 14 line 24. (The Examiner did in fact cite this part of Chen as being relevant to the second limitation (macro copied to a local document).)

Chen column 13 line 50 through column 14 line 24 is the passage of Chen that is closest to both of Applicant's aforesaid two critical limitations. In construing Chen, one should keep in mind that Chen's "template file" is part of what Applicant refers to as "global environment 13"; and that Chen's "application data file" is analogous to Applicant's "local document 11". Chen further states that a template file can have a .dot extension when the application is Microsoft Word, and an application data file can have a .doc extension when the application is Microsoft Word.

The passage at Chen column 13 line 50 through column 14 line 24 states that one way to detect the presence of a macro is to detect that combination of a "macro enablement instruction" and a "macro reproduction instruction". Chen defines a "macro enablement instruction" as one that formats a file to indicate that the file includes a macro for execution. Column 13 lines 52-54. This does not suggest the "move" nor the "global environment" in Applicant's first critical limitation (instructions causing a macro to be moved to a global environment). While Chen does state at column 14 line 9 that the instruction "FileSaveAs a\$,1" is one example of a macro enablement instruction, Chen defines this instruction to mean "keep an original file and save an additional copy of the file under a different format such as one that indicates that the file can include an embedded macro". Column 14 lines 9-12. There is no suggestion of a "move" or a "global environment" in this definition. While it is true that Applicant's claim 5 states

that a "SaveAs" command is one indicium of an instruction causing a macro to be moved to a global environment, Applicant's specification states that the "SaveAs" command is such an indicium in the special case where the code is written in Visual Basic. Page 12, lines 21-26. When Chen mentions the "FileSaveAs" command in column 14 line 9, he is talking about the application program Microsoft Word (column 13 line 64), which does not use Visual Basic for writing macros, but rather uses WordBasic (column 5 lines 37-38).

The second prong of Chen's test for the presence of a macro is the presence of a macro reproduction instruction. Column 13 lines 50-52. As an example of a macro reproduction instruction, Chen mentions the instruction "MacroCopy". Column 14 lines 16-20. Chen defines MacroCopy as "copies a macro, and if the macro is infected, all of its harmful instructions, from a source to a destination". Column 14 lines 16-20. This definition discloses the "copy" part of Applicant's second critical limitation (instructions causing a macro to be copied to a local document), but does not suggest the "local document" in said second critical limitation. Furthermore, "MacroCopy" is again a command in the WordBasic language used by Microsoft Word. Column 13 lines 64-66; column 5 lines 37-38. While it is true that Applicant's claim 6 states that a Copy command is one indicium of instructions causing a macro to be copied to a local document, Applicant's specification states that this is for the special case where the code is written in Visual Basic, not WordBasic. Page 13 lines 15-20.

The above discussion has demonstrated that Applicant's independent claims (claims 1, 13, 14, and 16) are patentable over the cited Chen reference. Applicant's remaining claims are dependent upon his independent claims, and therefore the patentability of these dependent claims follows from the patentability of the independent claims. Further with respect to claim 5, the SaveAs command is for the special case where the code is in Visual Basic, and further with respect to claim 6, the Copy command is for the special case where the code is written in Visual Basic, as discussed above.

Further with respect to claim 7, the passage of Chen (column 15 lines 15-41) cited by the Examiner does not suggest the concatenation operator recited in claim 7. Applicant's concatenation operator is described at page 18 lines 3-12 as a symbol, such as an ampersand, that concatenates two strings to assist detection module 17 in finding a location when the writer of malicious code has tried to obscure a location by breaking it into two pieces. The cited passage in Chen, on the other hand, pertains to examining two different instruction strings to see whether a macro is present. These strings could be, for example, the macro enablement instruction and the macro reproduction instruction described previously.

Further with respect to claim 8, the passage from Chen (column 14 lines 5-50) cited by the Examiner does not suggest Applicant's recitation in claim 8 that "each analyzing step makes substitutions for variable names when the code contains variable

names that are proxied". This recitation is discussed in Applicant's specification at page 18 line 13 through page 19 line 5.

Further with respect to claim 9, the passage of Chen cited by the Examiner (column 14 lines 37-57) does not suggest the recitation in Applicant's claim 9 that "each analyzing step traces the values of parameter variables when the code contains instructions that are invoked by other code". This recitation is discussed in Applicant's specification at page 19 lines 6 through 22.

Further with respect to claim 10, the passage of Chen (column 8 lines 2-3) cited by the Examiner does not suggest the recitation of claim 10 that "each analyzing step substitutes object names when the code is written in an object oriented programming language and when the code contains substituted object names". This recitation is discussed in Applicant's specification at page 19 line 23 through page 20 line 13.

For the above reasons, the Examiner is requested to withdraw her rejection of claims 1-16, and to allow these claims as written.

Applicant believes that this application is now in condition for allowance of all claims herein, claims 1-16 as written, and therefore an early Notice of Allowance is respectfully requested. If the Examiner disagrees or believes that, for any other reason, direct contact with Applicant's attorney would help advance the



Prosecution of this case to finality, she is invited to telephone the undersigned at the number given below.

Respectfully submitted,

Edward J. Radlo
Attorney for Applicant
Reg. No. 26,793

Fenwick & West LLP
Silicon Valley Center
801 California Street
Mountain View, CA 94041
(650) 335-7142

cc: SYMPOL
D. Chi.

20423/04318/DOCS/1411145.1